

**LAPORAN PRAKTIKUM**

**Bengkel Pemrograman Framework**

**Pengampu :**

**Maksum Rois Adin Saf, S.Kom., M.Eng.**

**Ardiyanto, S.Tr. Kom.**

Identitas :

No Absen : 12

Nama Lengkap : Mei Nanda Isnaini

NIM : 1955301072

Kelas : 2 TI C

**Judul Praktikum : Konsep MVC, Pemanggilan Library, Function**

**Hari/Tanggal : Rabu, 24 Maret 2021**

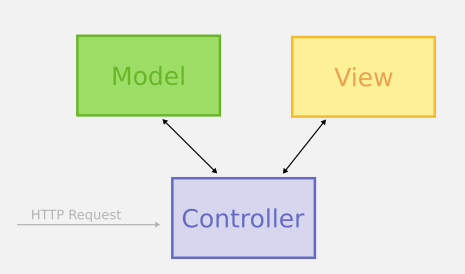
**Konsep MVC, Pemanggilan Library, Function**

1. **Dasar Teori**

MVC (Model, View, Controller) adalah sebuah pola desain (design pattern) arsitektur pengembangan aplikasi yang memisahkan dan mengelompokan beberapa kode sesuai degan fungsinya.

MVC membagi aplikasi ke dalam tiga bagian fungsional: model, view, dan controller.

* Model adalah kode-kode untuk model bisnis dan data. biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.
* View merupakan bagian yang menangani presentation logic. berisi kode-kode untuk tampilan.
* Controller merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.



Alur kerjanya seperti ini:

1. Mulai;
2. User mengirim request ke web;
3. File yang pertama kali dieksekusi adalah index.php;
4. Lalu dari index.php, request akan diteruskan oleh routers.php;
5. routers.php akan mencari cache di server, apabila tedapat cache maka cache itu yang akan dikirim sebagai balasan (response). Apabila tidak ada cache barulah request diteruskan ke *Controller*;
6. Controller akan bertanggunag jawab untuk mengambil data dari Model dan me-rendernya ke dalam View dengan menggunakan library, plugin, dan helper yang ada.
7. Hasil render (view) dikirim ke pengguna dan disimpan dalam cache, apabila fitur cache aktif;
8. Selesai.

Library adalah sekumpulan kelas dan fungsi yang dibuat untuk membantu pengembang aplikasi untuk dapat membangun aplikasi dengan lebih cepat dan lebih efisien. Pada umumnya saat kita membuat aplikasi web ada beberapa kelas yang hampir selalu digunakan, sehingga kelas-kelas tersebut dapat di atur supaya secara otomatis di-load oleh system dan dapat langsung digunakan.

Pada CodeIgniter library dibagi menjadi 2 yaitu library yang bersifat global dan library yang dapat dibuat sendiri sesuai kebutuhan. Library global terdiri dari kelas dan fungsi-fungsi yang telah disediakan oleh CodeIgniter, dan terletak pada folder system/libraries. Sedangkan library yang kita buat sendiri sesuai dengan kebutuhan ditempatkan pada folder application/libraries.

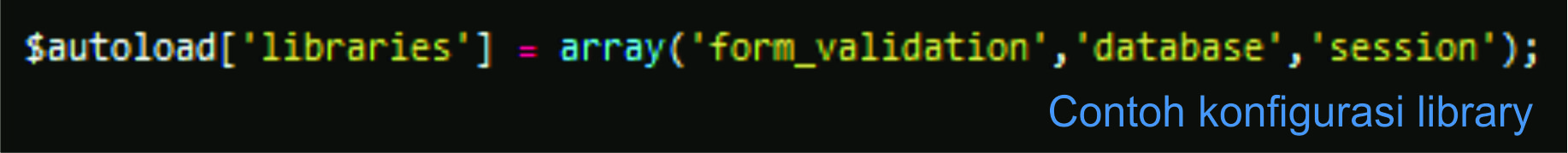
Beberapa library yang wajib diketahui oleh pengembang di antaranya adalah :

1. **Database**, library yang digunakan untuk mengakses database dan melakukan pengolahan data yang ada di dalam database. Database yang di dukung oleh CodeIgniter adalah mysql, mssql, oracle dan postgres. Sedangakan database yang tidak didukung secara langsung dapat dijembatani dengan driver odbc.
2. **Input**, library yang digunakan untuk menangani dan memproses data-data yang berasal dari form. Misalnya apabila kita menggunakan form untuk memasukan data maka library ini harus di-load supaya dapat melakukan pemrosesan data form.
3. **File Uploading**, library yang digunakan apabila kita akan membangun web yang dapat mengunggah (upload) file ke dalam web. Misalkan kita menginginkan supaya di dalam web kita ada fitur yang dapat digunakan untuk memasukkan file gambar ke dalam aplikasi web kita, maka digunakanlah library ini.
4. **Session**, library yang digunakan untuk memelihara informasi status mengenai pengguna. Sebagai contoh misalkan kita membangun suatu website dimana pengunjung website tersebut harus melakukan proses login terlebih dahulu untuk masuk ke dalam suatu halaman, maka pada situasi seperti ini, library session harus di-load supaya kita dapat memelihara state dari pengunjung, sampai pengunjung tersebut logout.
5. **URI Class**, library ini berisi fungsi-fungsi yang membantu kita untuk mendapatkan informasi dari URI pada alamat web kita.
6. **Validation**, library ini digunakan untuk melakukan validasi terhadap form input yang ada pada aplikasi web kita.
7. **Pagination**, library ini berguna pada saat kita memiliki banyak data yang harus ditampilkan. Misalkan kita memiliki 100 data, dimana ke-100 data ini akan ditampilkan ke dalam 10 halaman (10 data / halaman). Untuk membuat 10 halaman yang masing-masing memuat 10 data dan masing-masing halaman terhubung satu sama lain, maka pagination merupakan library yang tepat untuk digunakan.

Pada umumnya ada banya library yang dapat digunakan pada CodeIgniter. Tetapi untuk tahap awal library diatas yang wajib diketahui sebab library tersebut pada umumnya sering digunakan.

Untuk menggunakan library yang ada pada folder system/libraries, ada dua cara yang dapat dilakukan, yaitu :  
Mengatur pada file system/config/autoload.php.

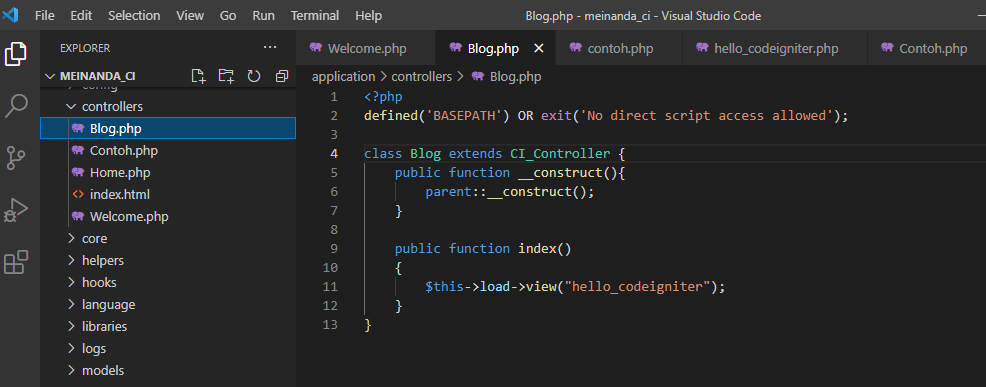
**Contoh :**

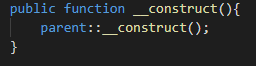


Dengan melakukan loading terhadap library yang kita inginkan pada controller dimana library ini akan digunakan. Biasanya library ini di-load pada konstruktor dari controller yang bersangkutan. Berikut sintaknya:

$this->load->library(‘nama\_library’);

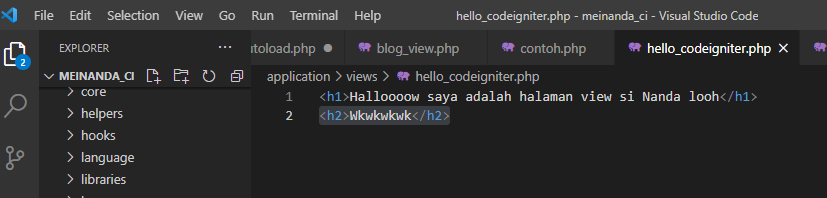
1. **Percobaan**
2. **Penerapan Controller dan View**
3. Membuat file controller baru dengan nama Blog.php pada application/controller/. Kemudian menambahkan kode program berikut :



Function \_\_construct()  merupakan constructor.

Terlihat pada function index() terdapat  yang mana berfungsi untuk meneruskan ke halaman hello\_codeigniter.php yang ada di Views.

1. Membuat file hello\_codeigniter.php pada application/view/. Kemudian menambahkan kode program berikut :

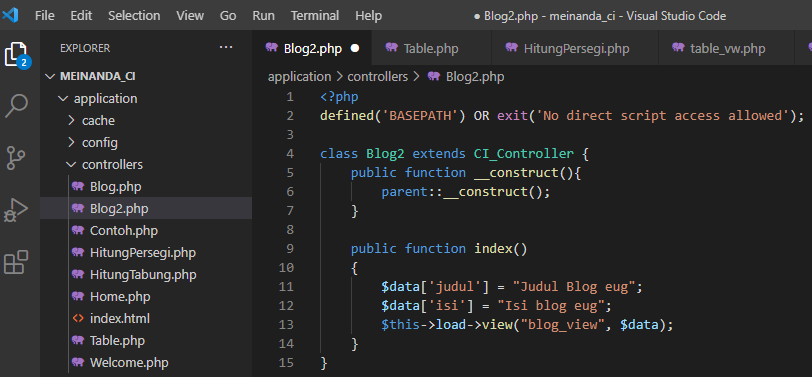


Saaat dijalankan dengan memanggil url <http://localhost/meinanda_ci/index.php/blog>, maka hasilnya seperti :

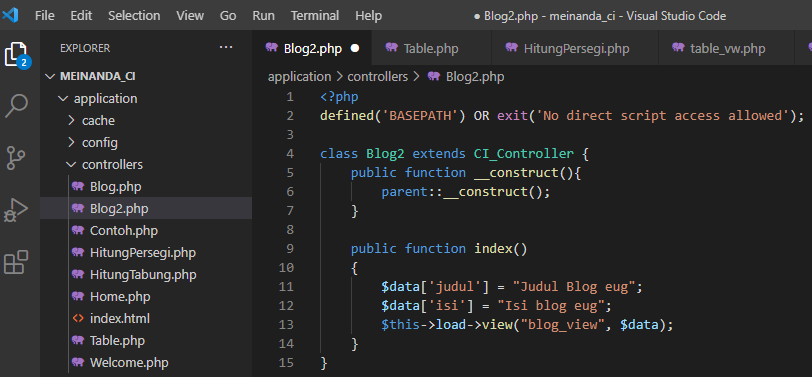


Fungsi view memiliki 3 parameter, yakni :

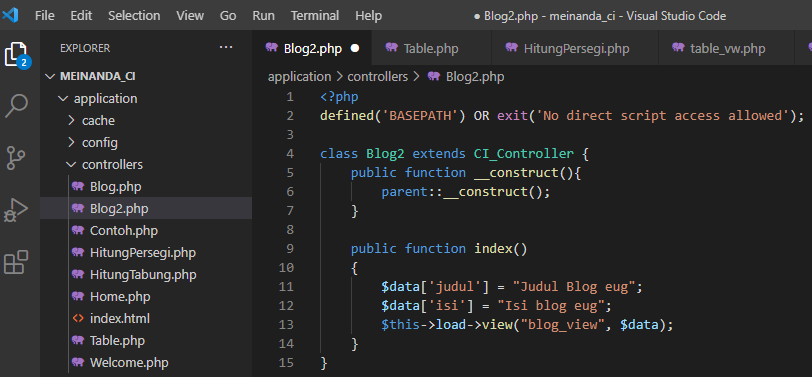
* Nama file view, file yang terletak di application/view yang akan di-load.
* Data parameter, digunakan untuk meneruskan data dari controller ke view.
* Misalnya dengan membuat sebuah file di controller dengan nama Blog2.php, lalu menuliskan kode program berikut :



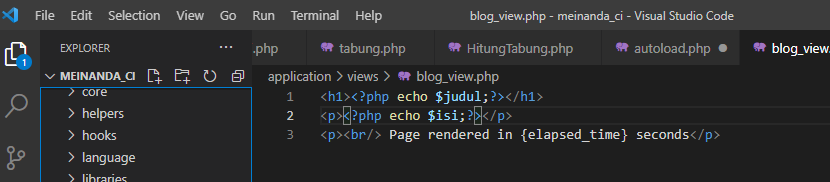
Terlihat pada line ke-11 sampai 12 terdapat variabel data yang menampung data judul dan juga isi :



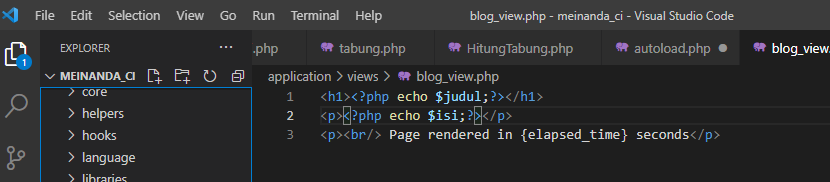
Sedangkan untuk pemanggilan data tersebut di file view dapat dilihat pada :



* Membuat file view dengan nama blog\_view.php, lalu menuliskan kode program berikut :

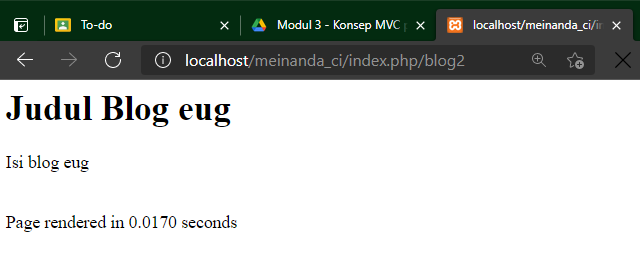


Sedangkan untuk mengakses isi dari variabel data cukup dengan pemanggilan isi variabel data yang ada pada file controllernya saja. Hal ini terlihat pada :

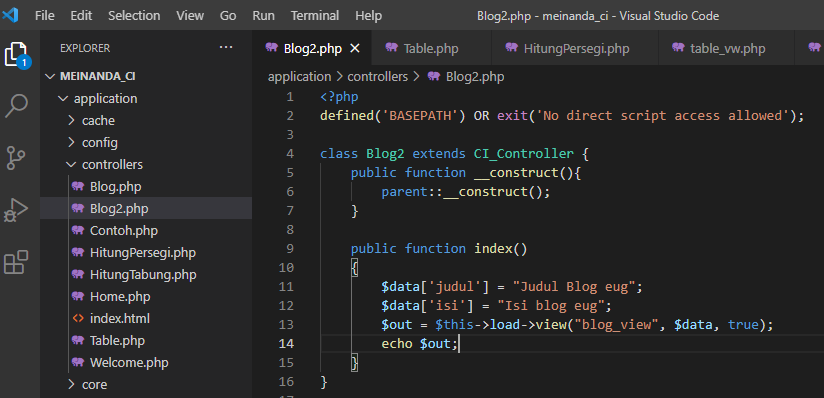


Yang mana $judul dan $isi merupakan isi dari variabel $data pada file controller.

Saat dijalankan hasilnya akan seperti :

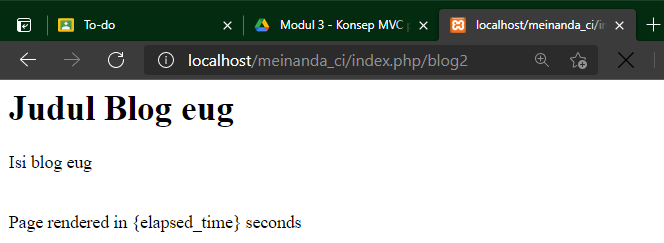


* Output parameter, parameter yang di set true jika kita ingin menyimpan hasil view ke dalam sebuah variabel. Contohnya :



Kode diatas berarti kita akan menyimpan hasil view kedalam sebuah variabel.

Saat dijalankan hasilnya akan seperti :

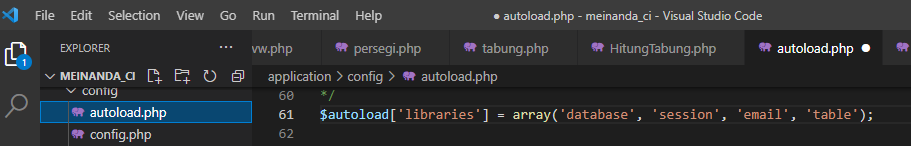


Terlihat dari hasil di web browser yang tadinya tertera waktu rendering, sedangkan pada kali ini tidak ditampilkan waktunya.

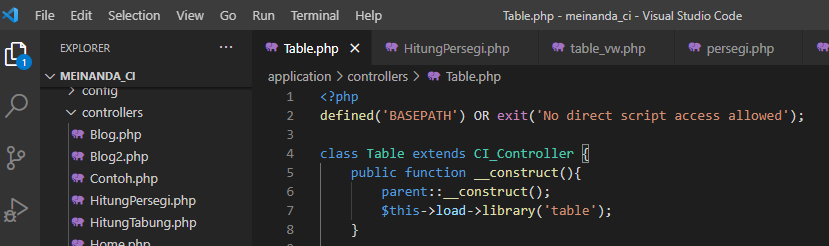
1. **Penggunaan Library pada CodeIgniter**

Terdapat 2 (dua) cara untuk memanggil library pada codeigniter, yakni :

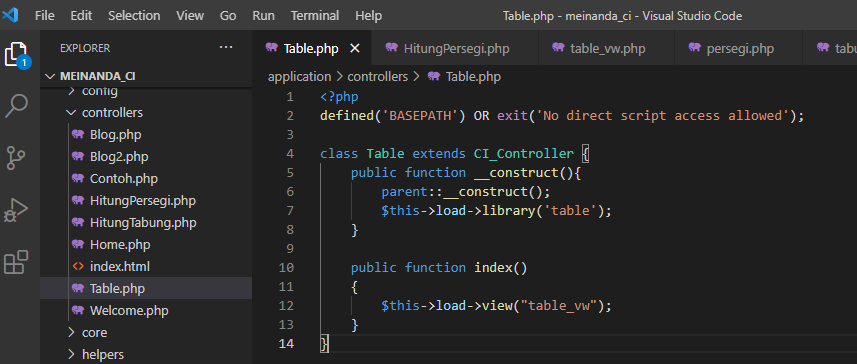
* Menambahkan library pada file application/config/autoload.php, seperti :



* Menambahkan pada constructor file yang terdapat di controller, seperti :

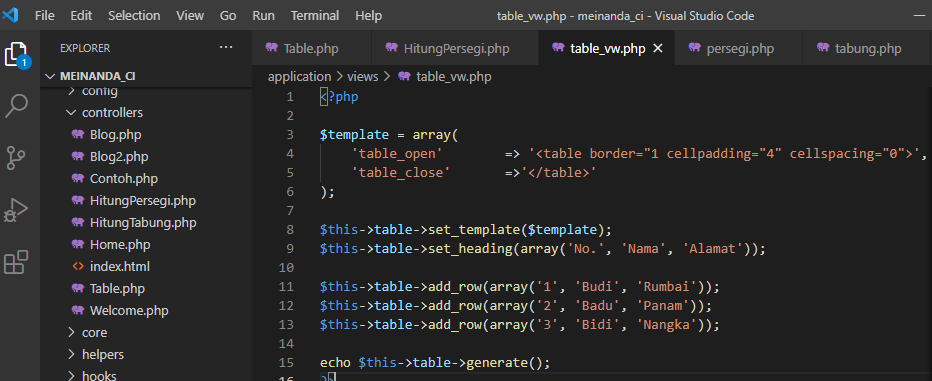


1. Membuat sebuah file controller dengan nama table.php, lalu menuliskan kode program berikut :



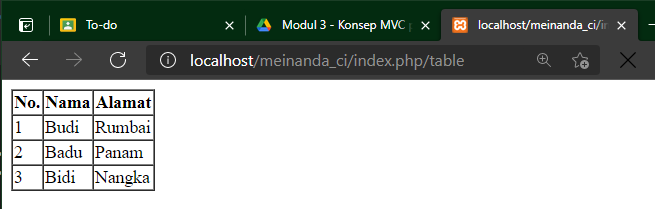
Dapat dilihat pada file di atas pemanggilan library diletakkan pada function \_\_construct yang mana akan mengambil data pada library yang ada pada CI.

1. Membuat halaman view dengan nama table\_vw.php pada application/views, lalu menuliskan kode program berikut :



$template digunakan untuk menentukan bentuk tabel yang akan ditampilkan di halaman web. Dan pada line ke-9 terlihat untuk mengatur heading pada tabel yang digunakan.

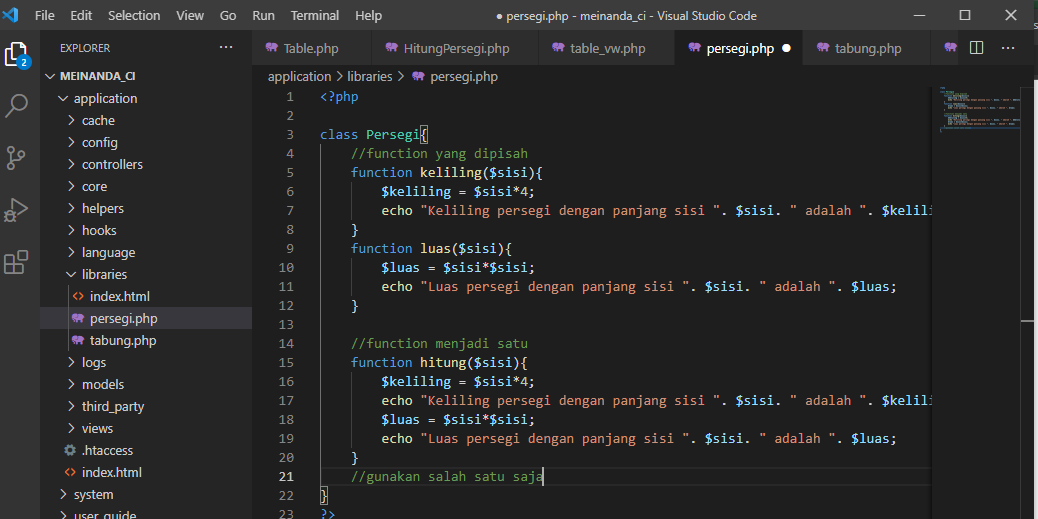
Saat dijalankan, maka hasilnya akan seperti :



1. **Membuat Library Sendiri**

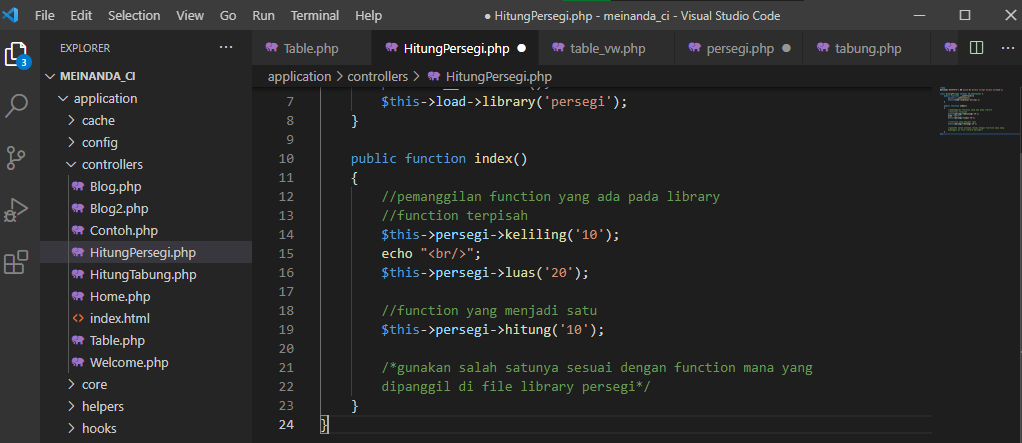
Untuk membuat library sendiri dapat dilakukan di folder application/libraries/ yang telah disediakan oleh codeigniter.

1. Membuat sebuah file dengan nama persegi.php pada application/libraries, lalu menuliskan kode program berikut :



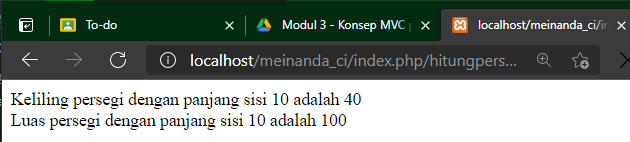
Saat membuat library sendiri sebenarnya tidak jauh berbeda dengan pembuatan class pada Java, dengan adanya class dan function (method pada Java).

1. Membuat sebuah file controller dengan nama HitungPersegi.php, setelahnya menambahkan kode program berikut :

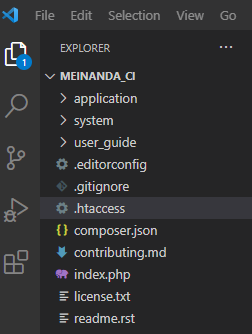


Pada file controller yang dipanggil untuk mengakses library ialah nama library dan nama function yang digunakan.

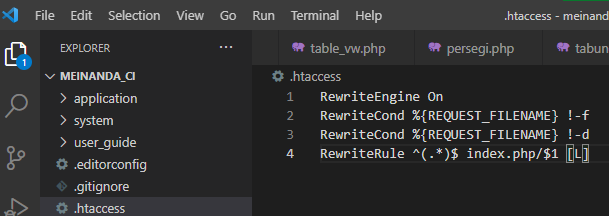
Memilih salah satu function yang akan digunakan (function hitung), maka pemanggilannya cukup  dan hasilnya saat dijalankan di web browser ialah :



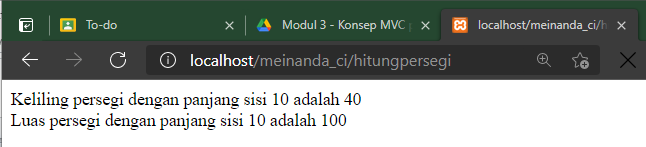
1. **Mempercantik URL CodeIgniter**
2. Untuk mempesingkat atau memudahkan kita dalam mengingat alamat url website kita, kita dapat melakukannya denga menambahkan file baru dengan nama .htaccess, penempatan file sejajar dengan folder application.

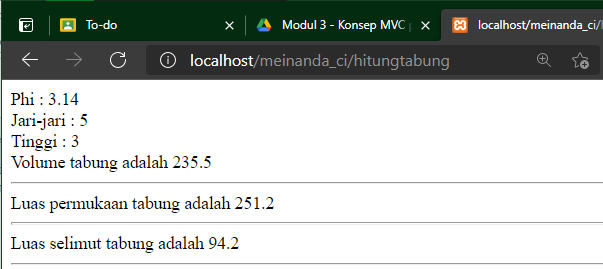


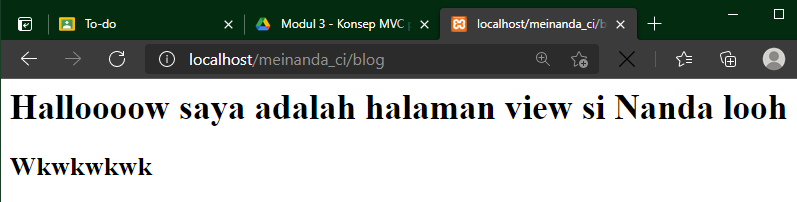
1. Lalu menambahkan kode program berikut :

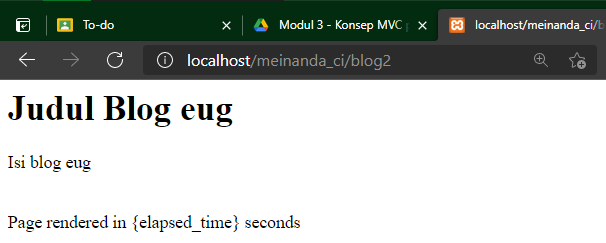


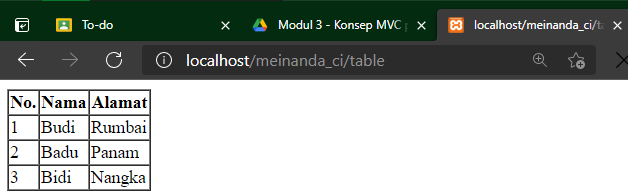
Maka saat hendak menjalankan url controller kita tidak perlu lagi menyertakan index.php, misalnya :







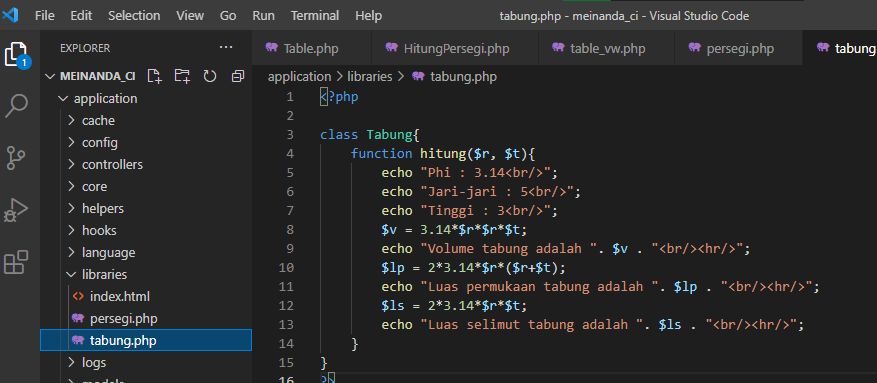




1. **Latihan**

**Membuat sebuah library baru untuk mengitung volume, luas permukaan dan luas selimut tabung.**

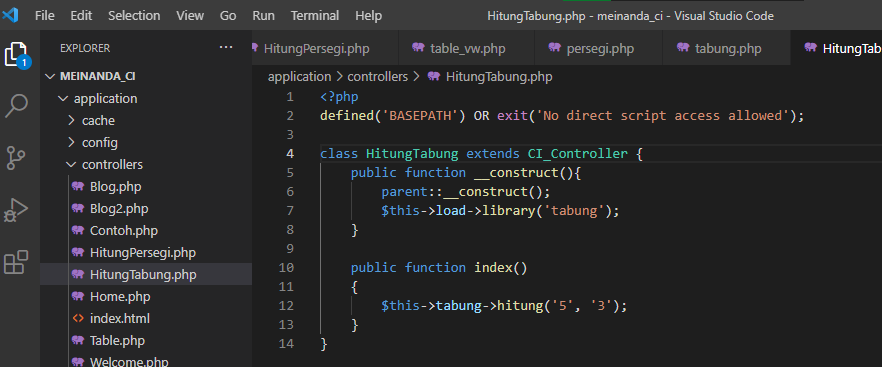
1. Membuat sebuah file di application/libraries dengan nama tabung.php, lalu menuliskan kode program berikut :



Pada library di atas, terdapat satu function yangmana terdiri dari rumus volume, luas permukaan dan luas selimut tabung, disertai outputnya.

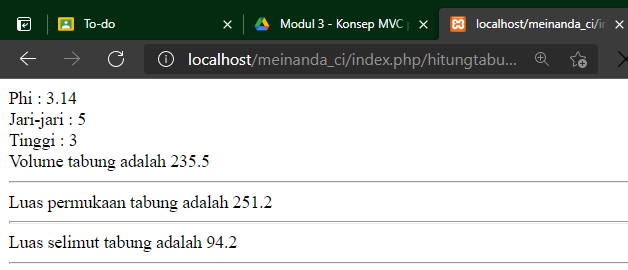
$v 🡪 untuk volumem $lp 🡪 untuk luas permukaan, $ls🡪 untuk luas selimut

1. Membuat sebuah file di controller dengan nama HitungTabung.php, lalu menuliskan kode program berikut :



Sehingga untuk pemanggilannya cukup dengan memanggil nama library (tabung) dan nama function yang akan digunakan (hitung).

Maka saat dijalankan hasilnya seperti :



1. **Daftar Pustaka**

<https://www.petanikode.com/codeigniter-mvc/>

<https://idcloudhost.com/panduan/mengenal-library-helper-dan-mempercantik-url-codeigniter/>